

CLAIMS

1. A processor capable of executing a secure hash algorithm (SHA),
comprising:

5 a core having a first execution unit and a second execution unit, wherein the
first execution unit is capable of processing a message and producing a partial result
passed to the second execution unit, the partial result capable of being processed by
the second execution unit in parallel with the processing of the message by the first
execution unit.

10 2. A processor capable of executing a secure hash algorithm (SHA) of claim
1, wherein the first execution unit is a single instruction multiple data (SIMD)
execution unit.

15 3. A processor capable of executing a secure hash algorithm (SHA) of claim
1, wherein the second execution unit is an integer execution unit.

4. A processor capable of executing a secure hash algorithm (SHA) of claim
1, wherein the message is a parsed padded message.

20 5. A processor capable of executing a secure hash algorithm (SHA) of claim
4, wherein the parsed padded message includes an original message and a plurality of
pad bits, the original message being a plurality of bits.

6. A processor capable of executing a secure hash algorithm (SHA) of claim 1, wherein the partial result includes a group of bits capable of being represented by a hexadecimal value.

5 7. A processor for cryptographic computation, comprising:
a first execution unit capable of performing a message schedule computation and producing a partial result, wherein the partial result includes a group of bits capable of being represented by a hexadecimal value; and
a second execution unit capable of performing a compression function using
10 the partial result, wherein the second execution unit is capable of operating in parallel with the first execution unit.

8. A processor for cryptographic computation of claim 7, wherein the first execution unit receives a plurality of blocks, the plurality of blocks including an
15 original message and a plurality of pad bits.

9. A processor for cryptographic computation of claim 8, wherein the message schedule computation includes a rotation operation capable of rotating the plurality of blocks.
20

10. A processor for cryptographic computation of claim 7, wherein the second execution unit includes an addition function capable of adding the partial result.

11. A processor for cryptographic computation of claim 7, wherein the partial result includes loosely coupled data capable of permitting parallel processing of the partial result in the first execution unit and the second execution unit.

5 12. A method, comprising:
receiving a message; and
performing a cryptographic computation on the message, the cryptographic
computation being capable of,
performing a hash computation such that the cryptographic
10 computation includes operations for,
performing a message schedule computation on
a first execution unit with a block of
data,
producing a partial result, and
15 performing a compression function on a second
execution unit with the partial result in
parallel with the message schedule
computation.

20 13. A method of claim 12, wherein the cryptographic computation is further
capable of performing a preprocessing operation.

14. A method of claim 13, wherein the preprocessing operation includes padding the message;
parsing a padded message; and
setting initial hash values.

5

15. A method of claim 12, wherein performing the message schedule computation further includes assigning rotated bits in the block of data to the partial result.

10 16. A method of claim 12, wherein performing the compression function further includes adding in the partial result.

17. A method for a one-way cryptographic hash computation, comprising:
processing a block in a first execution unit and producing a partial result;
15 sending the partial result to a second execution unit; and
processing the partial result in parallel with the first execution unit.

18. A method for a one-way cryptographic hash computation of claim 17,
wherein processing the block further includes rotating bits in the block, the bits in the
20 block capable of being represented as a hexadecimal value.

19. A method for a one-way cryptographic hash computation of claim 17,
wherein processing the partial result further includes rotating bits in the partial result,
the bits in the block capable of being represented as a hexadecimal value.

25

20. A computer program embodied on a computer readable medium for providing a cryptographic computation, comprising:

instructions for performing a hash computation using a first execution unit and a second execution unit, wherein the first execution unit partially produces a result for parallel processing by the second execution unit.

21. A computer program as recited in claim 20, wherein the instructions further rotate bits to produce the result.

22. A processor, comprising:

a first execution unit capable of performing a message schedule computation and producing a partial result, wherein the partial result includes loosely coupled data capable of permitting parallel processing; and

a second execution unit capable of performing a compression function consuming the partial result, wherein the second execution unit is capable of operating in parallel with the first execution unit.

23. A processor of claim 22, wherein the first execution unit includes at least one single instruction multiple data (SIMD) execution unit capable of transferring data to the second execution unit.

20

24. A processor of claim 22, wherein the second execution unit includes at least one instruction execution unit capable of transferring data to the first execution unit.

25. A processor of claim 23, further including a first register file capable of transferring data.

26. A processor of claim 25, further including a second register file capable of
5 transferring data.

27. A processor of claim 22, further including at least one instruction capable
of transferring data between the first execution unit and the second execution unit.

10